PRE Issue Tracking

$Revision: 1.1 $

$Date: Apr 08 1998 08:09:58 $

This document describes the goals, architectural design and aspects of implementation for issue tracking. Issue tracking is one of the configuration management areas. Issue tracking supports any of the system development or enabling activities (e.g. configuration management, verification and validation, project management) for the Product Realization Environment (PRE) project.

Prepared by:
Decision Support Systems Business Area
Departments 6531 and 6533
Sandia National Laboratories
P. O. Box 5800
Albuquerque, NM 87185-1138

Prepared for:
Product Realization Environment Project
Sandia National Laboratories
P. O. Box 5800
Albuquerque, NM 87185

Table of Contents

1. Introduction

This document is identified as the *PRE Issue Tracking*, date and version of this document.

During any of the system development and maintenance phases, an ***issue*** is a change request which may be:

- *problem or bug* - any deviation from expectations derived from specifications (requirements, design, interface), user documentation, standards or perhaps from someone's perception

- *enhancement* - a system change that is an addition to the specifications

Issue reporting is part of the change control aspect of configuration management, and [ieee-1042] indicates that a tool to support software change request/authorization tracking is part of an advanced tool set.

1.1 Goals

The goals of the issue tracking include

- assist in ***product assessment***
    - criticality
    - problem arrivals
    - problem density - number of problems related to some system area
- assist in ***project management***
    - making informed deployment decisions
    - tracking enhancements and problem fixes
- assist with **maintenance**
    - troubleshooting steps for past problems may be applied to current problems
- assist in ***testing***
    - issues are part of test results
    - regression test planning
- ***process improvement*** - where are problems occurring with respect to:
    - development process
    - development job (configuration management, test, development, project management)

The goals of the issue tracking system include

- supports multiple products within PRE
- supports different kinds of user communities
    - testers
    - developers
    - customers - end users
    - managers

1.2 Operational Principles

The following list outlines the operational principles for issue tracking for the PRE project. The PRE Configuration Control Board (CCB) manages reported issues. More details of the issue tracking processes are contained in section 0.

- Issues are reported to the issue tracking system (process) within _____ (time frame) after discovery.

- Issues are analyzed within _____ (time frame) after being reported.

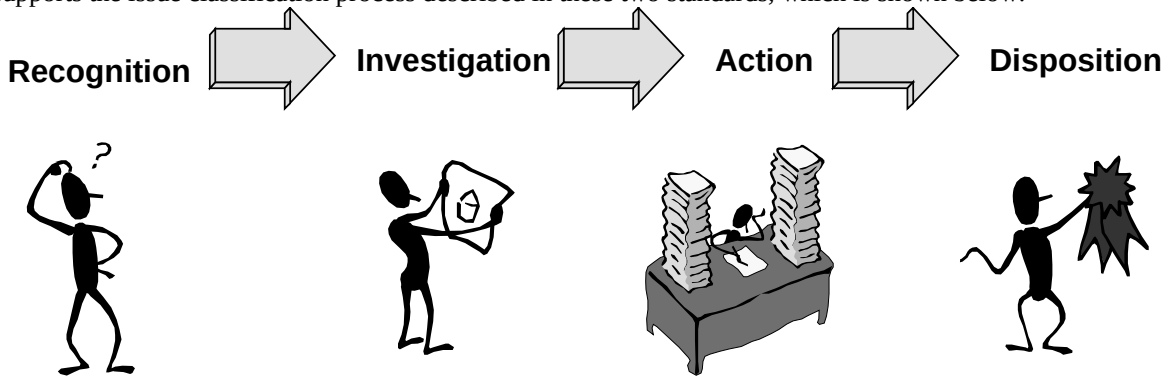
- Issues must be reviewed for priority by the CCB before work on the solution begins.
- Issues must be reviewed before closure.
- The solution to an issue is tested by an independent test team when required by the reviewers.
- An issue should include any relevant information that will be useful in troubleshooting and/or resolution. From [GNATS]: Richard Stallman writes, "The fundamental principle of reporting bugs usefully is this: report all the facts. If you are not sure whether to state a fact or leave it out, state it!" This holds true across all problem reporting systems, for computer software or social injustice or motorcycle maintenance. It is especially important in the software field due to the major differences seemingly insignificant changes can make (a changed variable, a missing semicolon, etc.).
- Each issue report should contain only one issue. This aids in tracking and analyzing each issue.
- Adhere to the standard format as much as possible. The greater the adherence, the less manual intervention required which is better spent working on the issue.
- The people who can change the issue state include: person responsible, CCB member or someone acting for CCB member.
- Notification shall occur as follows:
  - Submitter notified each time the issue state changes
  - Old and new responsible parties are notified when the responsible party changes
- Any further mail concerning an issue after submission should be carbon-copied to the GNATS mailing address with category and ID# in the subject line of the message. For example:

  `Subject: Re: PR category/gnats-id: original message subject`

  Such messages automatically go into the audit trail.
- Caution: Avoid deleting information from TEXT and MULTITEXT fields. (Refer to the table in section 0.)

1.3 Issue Classification Process

The ***issue tracking system design is based on*** [ieee1044] and [ieee1044-1] standards tailored for the PRE project. It supports the issue classification process described in these two standards, which is shown below.



There are several distinct, but related processes that correspond to the issue types discussed in section 0.

- bug tracking process
- enhancement request tracking process

The process diagrams for each issue type are described in the following subsections.

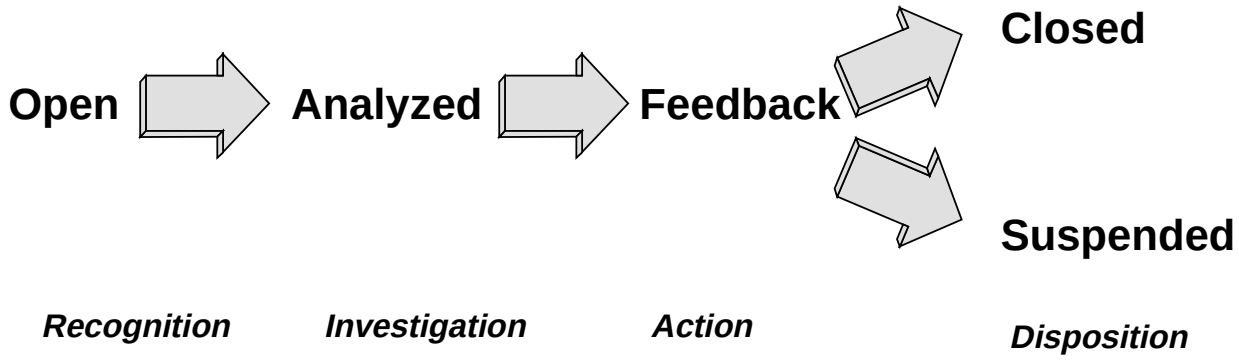
1.3.1 Bug Tracking Process Diagram

1.3.2 Enhancement Request Tracking Process Diagram

2. Detailed Design

2.1 Issue Life Cycle

The life cycle of an issue is shown below and is mapped to the IEEE issue classification process. The meaning of each state is contained in section 0.



2.2 Issue Attributes

To support the goals listed earlier in this document, attributes are used to help identify

- *when* a problem is found - development phase, activity, date/time
- *who* found the problem, who owns the problem
- *what* are the problem symptoms, cause
- *where* is the problem

The attributes are based on the issue process phases, i.e. when these attributes are first entered.

The attribute values, description, specification of when the field is required at what point in the submission process, and when the data entry fields appear on the GNATS submission screens is included in the following table. A field is one of ENUMERATED (specified set of field values), TEXT (single line of

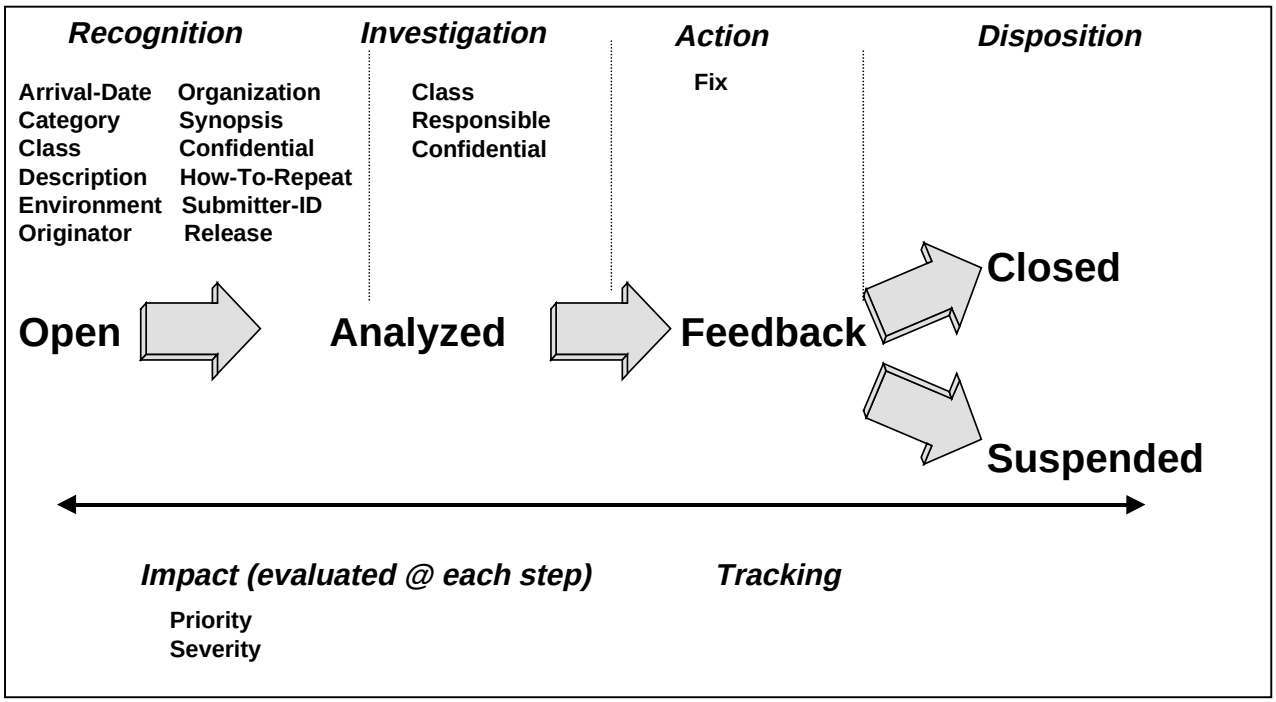
text), or MULTITEXT (any length text). Descriptions are adapted from [GNATS] and are listed in alphabetical order.

Attribute Name	Attribute Value(s)	Attribute Type	Description	Comments
Arrival-Date		TEXT	Time the issue was received by GNATS.	Field added by GNATS
Audit-Trail		MULTITEXT	Tracks changes in issue with respect to State, Responsible, and associated email.	
Category	df	TEXT – defined by PRE	Identifies the name of the product, component or problem where the problem lies. Refer to the categories file for the current list of categories and associated responsible people. Data Factory	Web submission screen Need to enrich with CM, test, vendor (maybe break out by layered products), spec, DB, process
	trader		Trader	
	fdk		Framework Development Kit	
	conversion		Conversion Central	
	unix-install		Unix installer	
	win32-install		Windows installer	
Class	sw-bug	ENUMERATED	General software product problem. Default value.	Web submission screen
	doc-bug		Problem with documentation.	
	change-request		Enhancement request.	
	support		Support problem or question.	
	mistaken		No problem, user error or misunderstanding. May indicate a problem indirectly in user documentation or training.	
	duplicate		This issue is determined to be a duplicate of a prior submitted issue.	
Description		MULTITEXT	Description of issue. Can attach files if needed.	Web submission screen
Environment		MULTITEXT	Description of environment where the problem occurred. Examples include, but are not limited to, machine architecture, operating system, host, libraries, pathnames.	Web submission screen
Fix		MULTITEXT	Specifies the change to address the issue.	
Number		ENUMERATED	Unique issue identifier. Included in the automated reply to the submitter, as well as the copy of the issue that's sent to the responsible person. Field is often paired with the Category field – category/number	
Originator		TEXT	Originator's name.	Web submission screen
Organization	world	MULTITEXT	Indicates the kind of issue. We want to track customer (world), developer (tech) and tester (QA) types of issues.	
	tech			
	qa			

Priority	high	ENUMERATED	High implies the issue is worked on before medium or low priority issues – as soon as possible.	Web submission screen
	medium		Medium means that the issue is worked on before low priority issues.	
	low		Low means this task is worked on if there are no outstanding medium or high priority issues.	
Release		ENUMERATED	Release or version number of the product, component or concept. Includes the release promotion state if a product or component.	Web submission screen
Responsible		TEXT	Person who is responsible for working on the issue.	Field added by GNATS
Severity	critical	ENUMERATED	Specifies the issue's impact. Should be reevaluated during each process phase. A critical severity implies the system or component crashes, hangs or loses data. No workaround exists. A critical problem will prevent a system or component from being released.	Web submission screen
	serious		Product doesn't work as advertised. A workaround may exist.	
	non-critical		Cosmetic problems.	
State	open	ENUMERATED	An open issue implies that it is either in the recognition process phase. The issue has been filed and the responsible person notified.	Field added by GNATS
	analyzed		The responsible person has analyzed the problem.	
	feedback		The problem has been solved, and the originator has been given information on the issue. The meaning of this state may differ for each issue process flow. Refer to the process flow chart.	
	closed		The changes have been reviewed, integrated, tested, documented and the issue has been confirmed that the solution works.	
	suspended		Work on the problem has been postponed.	
Synopsis		TEXT	Issue title.	Web submission screen
Unformatted		MULTITEXT	Any text that is not associated with the fields in the original issue.	
	Fields not currently on Web interface			
Confidential		ENUMERATED	Indicates this issue should be treated confidentially with limited distribution.	
How-To-Repeat		MULTITEXT	Specifies how to reproduce the problem. Provide sample code, input or steps. Maintainers can use sample code to test whether problem is fixed.	

Submitter-ID	TEXT	Used to specify site (e.g. Goodyear, EUVL, Diesel Collaboratory)	

The following figure illustrates the mapping of the fields onto the states and the IEEE issue tracking process.



3. Administrative Procedures

3.1 Tool Administration

[GNATS] contains information on tool administration. This section discusses details that are specific to the PRE project.

3.1.1 Utilities

[GNATS] lists the utilities that make up GNATS. Refer to [GNATS] for a complete explanation. To summarize, the utility categories are

- User utilities: Used to submit, query, update and view issue reports (send-pr, query-pr, edit-pr, view-pr respectively)
- Administrative utilities: Used by GNATS administrator
- Internal utilities: Tools used internally by GNATS – shouldn't have to invoke manually

3.1.2 Tool Administrator Duties

The duties of the GNATS administrator include:

- Configure GNATS for the support site (PRE)
- Edit issue reports GNATS cannot process
- Prune log files
- Set up new categories
- Back up GNATS database and associated configuration files
- Distribute send-pr so others may submit issue reports
- Run administrative and "internal" GNATS utilities as needed (mkcat, rmcat, gen-index,mkdist)

3.1.3 PRE Specific Definitions

Name	Description	Location
categories	Contains list of allowable categories.	
maintainers	???	
releases	Products against which submitters can log issues.	
responsible	List of people who can be responsible for an issue.	

3.1.3.1 GNATS Database Location

3.1.3.2 Environment Variables

The following table describes the environment variables used by GNATS.

Environment Variable Name	Description
NAME	Used to complete Originator field
ORGANIZATION	Used to complete Organization field
EDITOR	Used by send-pr to determine editor to invoke. vi is default.
PR-FORM	Filename of valid template.

3.2 Issue Reporting and Resolution

3.2.1 Who Reports Issues?

System developers, system testers (V&V[1] team or sanctioning team), project leads, selected system users and Corporate Computing Help Desk personnel will have access to the GNATS tool. Currently, anonymous/guest accounts for each project will not be supported because follow up on these issue reports is impossible.

3.2.2 Access Control

Authorization occurs at the project level due to the typically sensitive nature of the issue reports. Again, anonymous/guest accounts are not currently allowed.

3.2.3 Methods and Criteria for Reporting

3.2.3.1 Reporting Scope

There are two issue reporting scopes. These scopes are on a project basis, that is PRE.

1. Internal to V&V and development teams
2. External to the V&V and development teams

3.2.3.2 Tool

Issues will be reported and tracked using Cyclic's GNATS tool. References to more detailed analyses or test results should be contained in the issue report.

3.2.4 Issue Report Distribution and Timing

The issues are distributed via the GNATS tool.
The V&V team may opt to provide summaries of the issues being tracked on a recurring basis to:

- V&V project team
- Management team
- Development team
- Process owner

[1] From this point on, system testers and sanctioning team will be referred to as "V&V team".

The goal is to provide timely feedback to the development team. The periodicity should be negotiated between the different teams involved.
Customer-generated reports and enhancement requests should be distributed to the project lead of the development team as quickly as possible for analysis.

3.2.5 Methods and Criteria for Issue Resolution

[ieee-1059] ***requires*** *that each high criticality level anomaly shall be resolved satisfactorily before the V&V effort can formally proceed to the next life cycle phase*[2]. That requirement will not be followed rigidly because benefit can be derived from V&V activities in later phases, e.g. evaluation, irrespective of critical problems found in an earlier phase. Therefore, the V&V team will deviate from the standard's requirement.

3.2.5.1 Determination of Issue Severity and Impact Analysis

For issues discovered by V&V team members, the person who discovered the problem will initially assign a severity level and type. The team member may consult with other team members if unsure of the correct assignment. The test manager or project leader of that project has ultimate responsibility for reviewing issues prior to issue report distribution; the issues' severity will be adjusted if necessary.

3.2.5.2 Responding to Issue Reports

It is the *target*[3] *team's* responsibility to respond to issue reports. The project lead typically assigns an owner for each issue. The issue's owner then investigates and resolves the problem(s). The issue is closed by someone other than the owner who verifies the issue has been adequately addressed, e.g. development project lead or V&V test manager.

3.2.5.3 Authority for Response Evaluation

For the V&V team, response evaluation will be a shared responsibility. The V&V test manager for the project under test initially makes an initial assignment of the response, depending on relationship to the requirements areas in the V&V scope. The person assigned will evaluate the issue response. The evaluation may involve other tasks, e.g. regression testing of the affected code. The issue report will be considered resolved if the response reflects appropriate action taken, such as a software bug fix, and is verified by both the developer and V&V team.

3.2.5.4 Issue Tracking

The V&V team will be responsible for tracking the V&V team's generated issues. Issues tracked by the developers may be of interest to the V&V team as well and may be added as an externally produced issue if the team feels it is of high enough impact to warrant this action.

3.3 Task Iteration Policy

Software products that are input to a V&V task can change due to requirements changes or clarifications, or changes as a result of an enhancement/problem report. When changes are incorporated, some V&V tasks may need to be repeated, or new tasks introduced, depending on the modification. The extent to which the task needs to be repeated should consider the following issues:

- How fundamental is the change for the requirement area? Does it span requirement areas? The answer to this defines the scope of regression testing. If the change appears limited to the one requirement area, the task iteration may be limited to regression testing for the requirement, or a component that implements that requirement. If the change spans more than one requirement area, repeating an integration test in addition to the regression test may be more appropriate.

[2] [ieee-1059] pg. 46
[3] Any one of the system development jobs (project management, configuration management, V&V, development) or products can have an issue logged against it. For example, a test case problem is an issue for the V&V team; in this case, the target team is the V&V team.

- Will upstream and downstream products need to be re-evaluated due to the change?

3.3.1 Sources of Change Notification

The sources of change notices will be manually perused. Impending changes may be formally announced in one of the following ways:

- a design review
- GNATS issue report - problem or enhancement request

System behavior may be another source of change notification that wasn't documented for some reason.

3.3.2 Notifying Requirements Area Owners

The owners will notify the V&V team or be notified of potential change impacts via GNATS.

Appendix A: Acronyms and Abbreviations

CCB	Configuration Control Board
CVS	Version control system built on RCS, another version control system
PRE	Product Realization Environment
SOW	Statement of Work
UI	User Interface
V&V	Verification and Validation

Appendix B: References

Applicable Standards

[ieee-729] *IEEE Standard for Software Engineering Terminology*, IEEE standard 729-1983.
[ieee-829] *IEEE Standard for Software Test Documentation*, IEEE standard 829-1983.
[ieee-1012] *IEEE Standard for Software Verification and Validation Plans*, IEEE standard 1012-1986, revised 1992.
[ieee-1028] *IEEE Standard for Software Reviews and Audits*, IEEE standard 1028-1988.
[ieee-1220] *IEEE Trial-Use Standard for Application and Management of the Systems Engineering Process*, IEEE standard 1220-1994.
[ieee-1042] *IEEE Guide for Software Configuration Management*, IEEE standard 1042-1987.
[ieee-1044] *IEEE Standard for Classification of Software Anomalies*, IEEE standard 1044-1993.
[ieee-1044-1] *IEEE Guide to Classification of Software Anomalies*, IEEE standard 1044-1-1995.
[ieee-1059] *IEEE Guide for Software Verification and Validation Plans*, IEEE standard 1059-1993.

Vendor Tool Documentation

[GNATS] *Cyclic GNATS*, http://www.cyclic.com/cyclic-pages/GNATS.html.

Miscellaneous

[beiz] B. Beizer, *Software Testing Techniques*, 2nd edition, Van Nostrand Reinhold, 1990.
[rmp] *PRE Release Management Process.*